Beyond Patches: Mining Interpretable Part-Prototypes for Explainable AI

Mahdi Alehdaghi[1], Rajarshi Bhattacharya[1], Pourya Shamsolmoali[2], Rafael M. O. Cruz[1], Maguelonne Heritier[3], Eric Granger[1]

[1]LIVIA, Dept. of Systems Engineering, ETS Montreal, Canada
[2]Dept. of Computer Science, University of York, UK
[3]Genetec Inc.
{mahdi.alehdaghi.1, rajarshi.bhattacharya.1}@ens.etsmtl.ca, pshams55@gmail.com, mheritier@genetec.com,
{rafael.menelau-cruz, eric.granger}@etsmtl.ca

Abstract

As AI systems grow more capable, it becomes increasingly important that their decisions remain understandable and aligned with human expectations. A key challenge is the limited interpretability of deep learning models. Post-hoc methods like GradCAM offer heatmaps but provide limited conceptual insight, while prototype-based approaches offer example-based explanations yet often rely on rigid region selection and lack semantic consistency. To address these limitations, we introduce PCMNet, a part-prototypical concept mining network that learns human-comprehensible prototypes from semantically meaningful image regions without additional supervision. By clustering these prototypes into coherent concept groups and extracting concept activation vectors, PCMNet provides structured, concept-level explanations and enhances robustness to occlusion and challenging conditions, which are both critical for building reliable and aligned AI systems. Experiments on multiple image classification benchmarks show that PCMNet outperforms state-of-the-art methods in interpretability, stability, and robustness. This work contributes to AI alignment by enhancing transparency, controllability, and trustworthiness in AI systems.

Code — https://github.com/alehdaghi/PCMNet

Extended version — https://arxiv.org/pdf/2504.12197

Introduction

Deep learning (DL) has advanced computer vision, enabling models to achieve remarkable performance across a wide range of tasks, including object detection and image classification (Chai et al. 2021). However, despite this success, DL models often operate as black boxes, and their lack of transparency has raised critical concerns regarding interpretability and reliability. This issue becomes even more significant in areas such as healthcare (Fellous et al. 2019), self-driving cars (Kim and Joe 2022), and security systems (Chen et al. 2019; Nauta et al. 2023), where accurate predictions alone are insufficient; it is crucial to understand the reasoning behind them. In these situations, explainability is essential for building trust, diagnosing failures, and ensuring compliance with regulatory and ethical requirements (Markus et al. 2021). Consequently, there is growing interest in developing DL models that not only perform well but also provide interpretable and human-understandable explanations for their decisions (Saranya et al. 2023; Vilone et al. 2020).

Common post-hoc explainability methods, such as GradCAM (Chakraborty et al. 2022) and ScoreCAM (Wang et al. 2020), aim to explain a model's decision by identifying which parts of the input image had the most influence on the output. These methods analyze gradients or activations of trained models to identify salient regions of the input. An example is shown in Fig. 1 (a). Unlike these reverse-engineering approaches, ante-hoc methods are designed to make the model explainable from the outset. They guide the model to produce either global (He et al. 2025; Dreyer et al. 2024; Jiang et al. 2025) or local (Nauta et al. 2023; Ayoobi et al. 2025; Nauta et al. 2021; Zhu et al. 2024) interpretable parts that help explain its decisions. This idea is inspired by the recognition-by-components theory (Biederman 1987), which suggests that humans understand objects by decomposing them into meaningful parts or concepts.

Some methods, such as ProtoPNet (Chen et al. 2019) and PIPNet (Nauta et al. 2023), illustrated in Fig. 1 (b), divide the spatial deep features (prior to the final pooling layer) into fixed, small patches and learn a set of prototypes from these regions. These prototypes, activated by specific input patches, serve as interpretable visual features that support the model's decision. This also enables users to compare similar semantic concepts across different images, enhancing transparency and trust in the model's reasoning capacity.

However, these methods have notable limitations. Fixed patch sizes result in unstable prototype activation, particularly for larger semantic regions, where the patches fail to capture coherent or meaningful visual concepts. Conversely, very small patches lack sufficient contextual information, limiting their ability to represent interpretable or semantically rich components. Furthermore, discriminative and interpretable visual concepts often require larger regions to be represented clearly. When the model uses only small, fixed patches, it may fail to capture the full meaning of these concepts. This can lead to repetitive or shallow explanations, as seen in the bottom two rows of Fig. 1 (b), where similar patterns appear multiple times. To address this, the regions used for prototypes should not be fixed. Instead, they should be adjustable and learned during training, allowing the model to focus on more meaningful and diverse parts of the image.

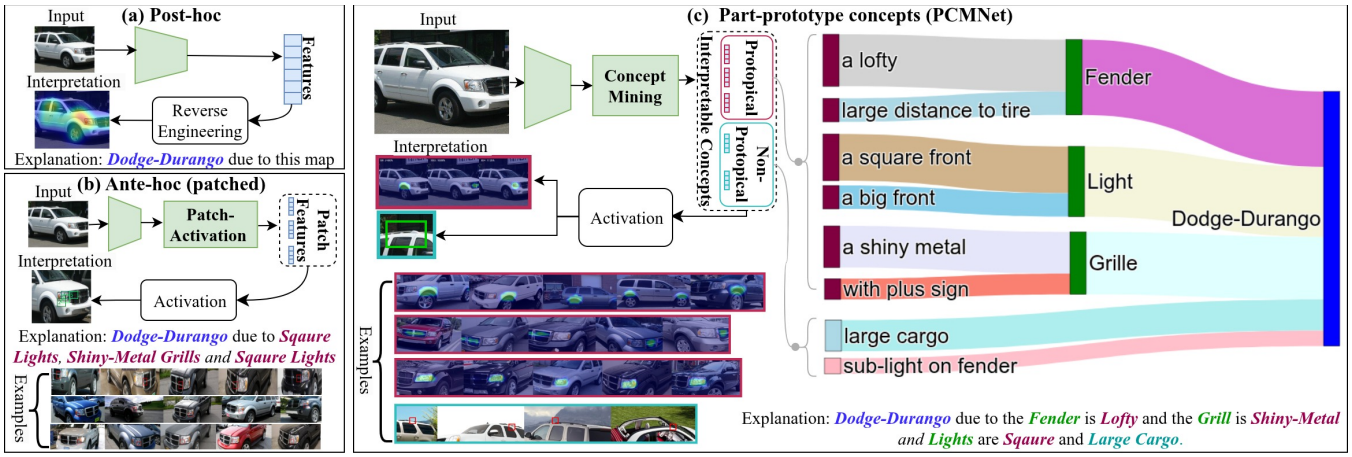


Figure 1: Explanation methods. (a) Post-hoc methods see the feature backbones as black-box models, and try to locate image regions activated by the most important features. (b) Ante-hoc (Patch-based) methods decompose the input into image patches and explain the decision using components derived with patches. (c) PCMNet, in contrast, mines both prototypical and non-prototypical concepts, offering a broader and more interpretable set of regions to explain model decisions.

One possible direction is to discover meaningful regions using unsupervised part-based or slot-based mechanisms. However, direct application of naïve part-discovery methods (van der Klis et al. 2023) or slot attention techniques (Monet et al. 2021; Locatello et al. 2020) often fails to produce semantically meaningful and interpretable components. These approaches typically aim to group similar visual parts across instances, but do not enforce conceptual consistency, resulting in ambiguous or fragmented representations (Rymarczyk et al. 2021). Moreover, Slot Attention-based features often lack explicit prototypical structures, making it difficult to associate them with clear and intuitive explanations (Li et al. 2021). As a result, the learned prototypes tend to lack diversity and interpretability, limiting the model ability to generate meaningful justifications for its predictions.

To address these limitations, we propose the Part-Prototypical Concept Mining Network (PCMNet), an interpretable image classifier that explains its predictions by extracting meaningful, part-based prototypical concepts[1] from selected image regions. PCMNet first identifies relationships between semantically related prototypes across different instances, even across class boundaries, and then derives a set of primitive class-discriminative concepts to form the basis for the model decision-making process.

In the first stage, PCMNet employs an unsupervised part-discovery module that learns prototypes through a novel center-clustering loss. In the second stage, deep features are encoded into a sparse Concept Activation Vector (CAV) (Kim et al. 2018), which provides a spatially and semantically interpretable representation of the model's decision. Each concept activation is computed as the cosine similarity between part features and class-aware prototype centroids. To ensure both conceptual simplicity and class-specific relevance, PCMNet applies a clustering algorithm to the part features associated with each prototype and abstracts each cluster into a representative concept vector. The Concept Activation Vector (CAV) produced by PCMNet is inherently interpretable, as it is generated through a sparse, non-negative linear transformation that ensures all concept scores are positive. As illustrated in Fig. 1 (c), PCMNet activates concepts such as "lofty fenders," "shiny metal grille," and "square-shaped lights" to support its prediction that the vehicle is a Dodge Durango, thereby providing a clear and meaningful explanation of the model's reasoning.

The contributions of this paper are summarized as follows. **(1)** We propose a part-prototypical concept mining network (PCMNet) that jointly learns adaptive semantic regions and class-discriminative prototypes for interpretable image classification. **(2)** A contrastive prototype learning mechanism is introduced that extracts semantically coherent concepts from adaptively sized image regions rather than fixed patches. **(3)** We develop a two-level clustering approach with pixel-level part discovery and feature-level concept formation. **(4)** PCMNet outperforms ProtoPNet and PIPNet in occlusion robustness (+7.2% accuracy at 30% occlusion) and explainability across multiple datasets.

Related Work

(a) Post-Hoc or Heatmap-Based Explanation. Heatmap-based explainability methods are a widely used family of post-hoc techniques that visualize which parts of an input image contribute most to a model's decision. These techniques, often referred to as attribution methods, assign importance scores to different image regions to highlight influential features. Gradient-based methods, such as GradCAM (Selvaraju et al. 2017) and ScoreCAM (Wang et al. 2020), generate heatmaps by backpropagating gradients concerning input features, revealing class-specific activation regions. While GradCAM produces class-dependent heatmaps, other methods like FullGrad (Srinivas and Fleuret

[1]We define **part-prototypes** as localized visual regions that commonly appear across inputs (e.g., light of a car), while the term **concept** describes a specific attribute or variation of that part (e.g., *rectangular* or *circular* shapes of the car light).

2019) are class-agnostic, providing a more general view of model behavior across different outputs. Despite their popularity, gradient-based methods suffer from high sensitivity to noise, which can lead to unreliable and inconsistent heatmaps. To mitigate this, gradient-free CAM techniques, such as ScoreCAM, have been proposed to generate more stable and interpretable explanation maps.

Beyond gradient-based methods, attribution propagation approaches offer an alternative way to decompose model predictions into layer-wise relevance scores. Techniques such as Layer-wise Relevance Propagation (LRP) (Bach et al. 2015) or KD-FMV (Jiang et al. 2025) recursively distribute relevance through the network, providing a structured breakdown of model decisions. While initially designed for convolutional neural networks (CNNs), recent adaptations have extended these methods to vision transformers (ViTs) (Chefer et al. 2021), using their self-attention mechanisms for improved interpretability. However, despite their effectiveness in highlighting important image regions, both gradient-based and attribution propagation methods remain fundamentally limited in providing high-level, human interpretable concepts. Unlike our PCMNet, which explains decisions through part-based prototypes, heatmap-based methods do not inherently capture semantic relationships between features, making them less suitable for interpretable concept-based explanations.

(b) Concept-Based Explanation. Concept-based XAI methods (Dreyer et al. 2024; Chen et al. 2019; Rymarczyk et al. 2021; Bach et al. 2015; He et al. 2025) analyze the role of latent representations in specific layers of a deep neural network to explore concepts from input images and find Concept Activation Vectors (CAVs) (Kim et al. 2018) for the made decision. Early XAI research primarily focused on understanding how these concepts contribute to global decision-making, identifying the most relevant concepts for a given output class (Dreyer et al. 2024; Bach et al. 2015). Some approaches enhance model interpretability by employing local feature attribution techniques to localize and quantify the importance of concepts for individual predictions, thereby enabling concept-based explanations at the instance level (Nauta et al. 2023, 2021; Ayoobi et al. 2025). While these methods provide faithful explanations aligned with the model's internal reasoning, they are vulnerable to performance degradation under occlusion, as their extracted concepts often rely on limited and highly localized discriminative regions of the input. PCMNet addresses this limitation by wider range of part-based prototypical concept reasoning to rely on other discriminative parts that are not occluded.

Recently, Concept Bottleneck Models (CBMs) has emerged as a research avenue for inherently interpretable approaches that predict intermediate human-understandable concepts before class prediction (Xie et al. 2025; Yuksekgonul, Wang, and Zou 2023; Rao et al. 2024; He et al. 2025). Unlike patch- or part-prototype models, CBMs explicitly separate the reasoning process into concept prediction and concept-to-label mapping, allowing direct human intervention on the concept layer. However, their reliance on annotated concepts limits scalability, and the fidelity of the bottleneck depends on concept quality. Recent methods, including post-hoc CBMs (He et al. 2025; Xie et al. 2025) and vision-to-concept tokenizers that learn visual concept vocabularies without text supervision—aim to bridge this gap by enabling scalable and more interpretable reasoning.

Prototypical Concept Mining Network

To explain classification decisions over a training set $\mathcal{T} = (x^1, y^1), ..., (x^N, y^N) \subset \mathcal{X} \times \mathcal{Y}$ with L classes, PCMNet extracts sparse and primitive concepts that are both activated and discriminative for object recognition. These concepts are captured through a compact set of prototypes that recur across images in the dataset (e.g., lights, fenders, and grilles for cars; heads and wings for birds). The model outputs a class-based Concept Activation Vector (CAV) $\mathbf{z}^i = [\mathbf{z}_1^i; \ldots; \mathbf{z}_K^i]$, where ; denotes concatenation and each $\mathbf{z}_p$ represents the activated concepts from part p.

Fig. 2 shows the overall overview of PCMNet, which transforms features extracted from the backbone into explainable and interpretable concepts in three steps. **Step 1**: PCMNet identifies semantically meaningful and independent regions from input images in an unsupervised manner by minimizing the Marginal Cluster Center loss ($\mathcal{L}_{\text{mcc}}$). This process learns part-prototype features $\mathbf{f}_p^i$ that are informative for class discrimination. **Step 2**: Once part-level features are obtained, the Concept Mining Module (CMM) aggregates these features across all instances of each class to discover a set of shared elementary visual patterns. Cluster centers derived from this process form the list $\mathcal{C}$, representing the core prototypical concepts. **Step 3**: For each input, a Concept Activation Vector (CAV) $\mathbf{z}^i$ is computed by measuring the similarity between its extracted part features and the concept list $\mathcal{C}$. These CAVs are used both for classification and for explaining the model's decision. To preserve class-specific but non-shared discriminative cues, PCMNet also incorporates features outside the prototypical regions into the CAVs.

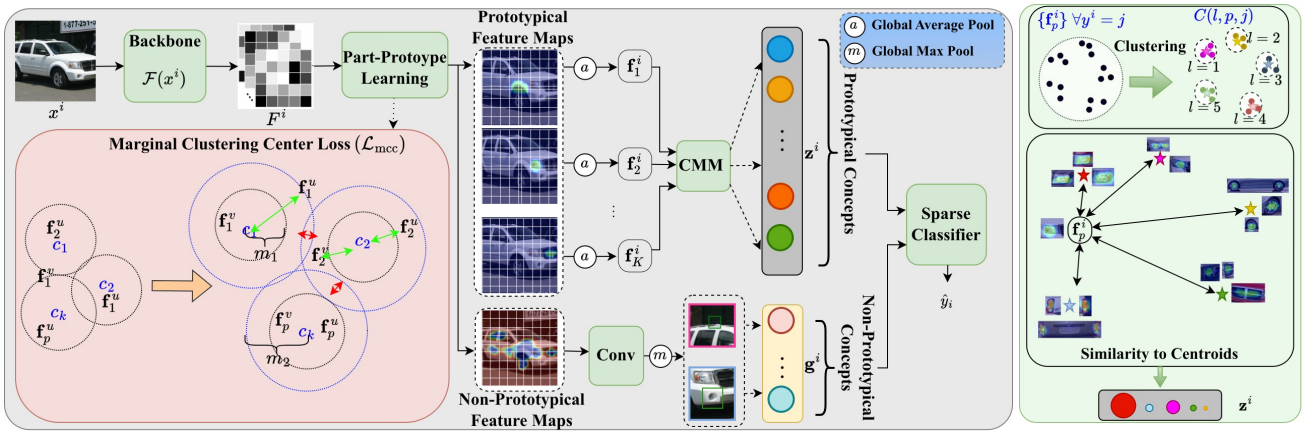
Step 1: Part-Prototype Discovery

Our model is shown in Fig. 2. The input images are given to a backbone to extract the 3D deep features as $F^i \in \mathbb{R}^{D \times w \times h}$ where D is the channel size and w, h denote the width and height. Based on extracted deep features F^i, our model finds semantically independent meaningful regions that can be shared in different inputs. We name them "part-prototype". For determining regions, the model scores each pixel in F^i to specify their belongingness to each part-prototype class. The mapping score for each pixel (a) is represented by $M_p^i(a)$, we have $\sum_{p=1}^{K+1} M_p^i(a) = 1$. The spatial feature for each part, F_p^i, is computed by element-wise multiplication of M_p^i to F^i. Then part features, $\mathbf{f}_p^i$ is computed as:

$$\mathbf{f}_p^i = \mathcal{GP}(\texttt{conv}(F_p^i)) \in \mathbb{R}^{d_f}, \tag{1}$$

where $\mathcal{GP}$ is the global average pooling and d_f is the dimension of features. The last index (K+1) is used to cover the background or non-part regions of the foreground (Alehdaghi et al. 2025). We name these features non-prototypical concepts that are computed as:

$$\mathbf{g}^i = \mathcal{MP}(\texttt{conv}(F_{K+1}^i)) \in \mathbb{R}^{d_f}, \tag{2}$$



(a) PCMNet (b) CMM

Figure 2: (a) Overall architecture of PCMNet. In the first stage, part prototypes are learned from spatial features extracted by the backbone. In the second stage, prototypes are clustered within each class to identify frequently occurring concepts. The center of these clusters is computed as concept prototypes, and the distance between part prototypes and concept clusters determines the activated concepts for the model's decision-making. This approach enables interpretable and concept-driven classification. (b) Concept Mining Module. First, a part centroid list is generated by applying DBSCAN (Ester et al. 1996) clustering within each class and computing the centers of the resulting clusters. These centroids serve as frequently occurring concept prototypes. Next, to activate the most relevant concept for a given part feature, the similarity (inverse distance) to the centroids is measured. These values are then used as CAVs to inform the model's decision-making.

where $\mathcal{MP}$ is the max-pooling. We use the PDiscoNet (van der Klis et al. 2023) model for extracting parts features. Our part loss comprises their final part-discovery loss ($\mathcal{L}_{\text{pd}}$) without their orthogonal component and with our marginal cluster center loss ($\mathcal{L}_{\text{mcc}}$):

$$\mathcal{L}_{\text{part}} = \mathcal{L}_{\text{bl}} + \alpha \mathcal{L}_{\text{mcc}}. \tag{3}$$

where α is a parameter regularizing the effect of $\mathcal{L}_{\text{mcc}}$. The goal of $\mathcal{L}_{\text{mcc}}$ is to ensure that each selected region is consistent across inputs and captures semantically similar content, while remaining class-discriminative and identity-aware.

Marginal Cluster Center Loss To ensure part-prototypes are both distinct and semantically meaningful, we encourage separation in the feature space while guiding part-level features to align with their corresponding prototypes. To handle intra-part variation across classes, we apply a soft-margin loss that allows features from the same part (but different classes) to stay within a bounded distance of their prototype. This encourages the model to discover consistent, class-aware part structures while preserving inter-class separation between prototypes. The loss can be written as

$$\mathcal{L}_{\text{mcc}} = \sum_{p=1}^{K} \left([|\mathbf{f}_p^i - c_p| - m_1]_\star + \frac{1}{K} \sum_{q \neq p}^{K} [m_2 - |c_p - c_q|]_\star \right), \tag{4}$$

where c_p is a learnable prototype and $[\cdot]_\star$ is $\max(\cdot, 0)$.

Step 2: Part-Centroid List Generation

In **Step 1**, the backbone learns to extract meaningful part-level features that support the classification task. To discover compact and consistent class-specific concepts, we collect all part features from the training set. Specifically, for each class j and part p, we construct $B_p^j = \{\mathbf{f}_p^i \mid y^i = j\} \in \mathbb{R}^{N_j \times d_f}$, where N_j is the number of training samples from class j. To identify semantically coherent patterns, we apply the clustering algorithm separately to each B_p^j, yielding $T_p^j \in \mathbb{N}^{N_j}$, which assigns cluster labels to the part features. We then compute the centroid of each cluster l (representing a visual concept) as:

$$\mathcal{C}(l, p, j) = \text{mean}(\{B_p^j[T_p^j = l]\}) \in \mathbb{R}^{d_f}. \tag{5}$$

The total number of discovered concepts across all classes and parts is computed as $d_c = \sum_{j=1}^{L} \sum_{p=1}^{K} \max(T_p^j)$. This process, shown in the top of Fig. 2b, allows the model to build a compact and expressive vocabulary of class-specific, part-aware visual concepts.

Step 3: Concept Activation Mining

Once the centroid list is generated, each concept's activation value is defined by the similarity of extracted features to each centroid:

$$\mathbf{z}^i = \{\mathcal{S}(\mathcal{C}(l, p, j), \mathbf{f}_p^i) \; \forall p \in \{1..K\} \text{ and } j \in \{1..L\}\}, \tag{6}$$

where $\mathbf{z}^i \in \mathbb{R}^{d_c}$ ans $\mathcal{S}$ is cosine distance. We aim to explain each output class of our model using a small set of interpretable concepts. We, therefore, train a sparse classifier on top of z^i to obtain the final classification scores $o^i = W_1^T \mathbf{z}^i + W_2^T \mathbf{g}^i + b$ and the predicted class $\hat{y}^i = \arg\max(o^i)$. Here, $W_1 \in \mathbb{R}^{d_c \times L}$, $W_2 \in \mathbb{R}^{d_f \times L}$ and $b \in \mathbb{R}^L$ denote the classification weights and bias term, respectively. This sparse layer is trained with the following sparse classification loss (Wong, Santurkar, and Madry 2021):

$$\mathcal{L}_{\text{cl}} = \mathcal{L}_{ce}(W_1^T \mathbf{z}^i + W_2^T \mathbf{g}^i + b, y^i) + \lambda \mathcal{R}(W_1), \tag{7}$$

where $\mathcal{L}_{\text{ce}}$ is the cross-entropy loss, y^i is the label, λ is the sparsity regularization strength and $\mathcal{R}(W) = (1-\gamma)\frac{1}{2}\|W\|_F + \gamma\|W\|_1$, here $\|W\|_F$ is the Forbenius norm and $\|W\|_1$ denotes the element-wise matrix.

Overall Training

We jointly optimize the network in an end-to-end manner:

$$\mathcal{L} = \mathcal{L}_{\text{part}} + \beta\mathcal{L}_{\text{cl}}, \tag{8}$$

where β is hyperparameter. We set $\beta = 0$ during initial training and $\beta = 2$ after learning part-prototypical concepts.

Results and Discussion

Implementation Details

PCMNet is evaluated on two standard prototype-learning benchmarks: Stanford Cars (Krause et al. 2013) (196 car classes) and CUB-200-2011 (Wah et al. 2011) (200 bird species). We use a pretrained ResNet50 as the backbone. Each training batch (32 images) is resized to 488×488 with random cropping and padding. Optimization is performed using Adam (Kingma and Ba 2014). Regularization parameters λ and γ follow (Oikarinen et al. 2023); we set $\alpha = 1.5$, $\beta = 2$, with margins $m_1 = 0.3$ and $m_2 = 1.5$ based on hyperparameter tuning analysis in the extended version (Appendix A) alongside additional implementation details.

End-to-End Multi-Stage Training: PCMNet modules (part discovery, concept clustering, and concept activation) are trained end-to-end in a unified pipeline over 40 epochs:

- *Step 1* (initial epochs): The part discovery module is optimized to localize semantically meaningful part features. This stage continues until convergence of $\mathcal{L}_{\text{part}}$.
- *Step 2* (intermediate epochs): Concept clustering runs intermittently (every 5 epochs) to reduce computation while keeping centroids up to date with changing representations.
- *Step 3* (final epochs): The Concept Activation Mining Module (CMM) is trained alongside the backbone, using the updated centroids from Stage 2 to align activated features with discriminative concept prototypes. At inference, PCMNet behaves as a standard forward-pass classifier.

Explainability Metrics

To evaluate explainability, we use established metrics from (Dreyer et al. 2024), with minor adjustments.

(a) Faithfulness: Faithfulness (Dasgupta et al. 2022; Chan et al. 2022) evaluates how much the activated concepts influence the model's final prediction, serving as a key metric for assessing the quality of explanations. One widely used approach for measuring faithfulness is concept deletion, which involves removing selected concepts from the model's reasoning process and observing the resulting change in output confidence. Prior works, such as (Dreyer et al. 2024), typically assess this by removing only the single most important concept. However, this limited scope fails to capture the broader contribution of other activated concepts and may not reveal the full extent of their impact. To provide a more comprehensive assessment, we extend this approach by successively removing the top-k most activated concepts and measuring the resulting drop in model confidence. A larger drop indicates higher faithfulness, as it suggests that the removed concepts were indeed critical to the model's decision. This extended evaluation allows us to better quantify how much the explanation aligns with the model's internal reasoning.

(b) Stability: The extracted concepts from input images that share some semantic concepts must be stable and explainable for unseen images. To evaluate stability, we compute CAVs on k-fold subsets of the data (k = 10 as default) similar to (Dreyer et al. 2024) for all classes at steps 2 and 3. We then map CAVs together using a Hungarian loss function and measure the cosine similarity between them.

(c) Consistency: Activated concepts from images with the same class should be similar to be consistent and explain the model's decision. To measure this consistency, we extract the CAV from images and then measure the cosine similarity between them. The mean cosine similarity is assessed between images from the same class and between images from different classes. The ratio between inter/intra-class similarity should be high for a stable explanation.

(d) Sparseness: The sparseness metric essentially describes the uniformity of the concept activation, where having some concepts activate more than others is considered easier to interpret. This is because a uniform distribution of concept activations would provide little information on the importance of specific concepts or parts of the images, i.e., a high entropy in the generated concepts.

Comparison to Prototype xAI Methods

To evaluate PCMNet, we report classification accuracy and XAI metrics, and compare them with other ante-hoc methods (ProtoPNet(Chen et al. 2019), PIPNet(Nauta et al. 2023), ProtoViT (Ma et al. 2024) and MCPNet (Wang, Wang, and Chiu 2024)) in Table 1. The consistency metric is reported in two forms: Intra and Inter, measuring the cosine similarity between activated concepts within the same class and across different classes, respectively. Faithfulness is reported as **F(*n*)**, measuring accuracy drops when the top ***n*** important concepts are removed from the decision. To compute the contribution of each concept to the final decision, we compute a weighted score as the product of its raw value and its corresponding classification weight.

Ablation Studies

(a) Effect of Number of Concept Center centroids (d_c): The size of the concept list (d_c) determines the semantic resolution of PCMNet and is influenced by both the number of parts and the number of training classes. As the number of parts (K) increases, so does the total number of extracted centroids. We analyze this dependency further in Appendix B. While richer concept sets can improve interpretability, they also increase memory and computational cost.

To manage this, we apply a post-clustering refinement step that merges similar centroids within or across classes using hierarchical clustering (Murtagh and Legendre 2011). Table 2 reports the effect of varying the merging level and threshold—defined as a percentage of the maximum inter-centroid distance—on the Cars dataset. We find that moderate merging (e.g., Level 1 with 10% threshold) significantly reduces d_c with minimal impact on classification accuracy or

	Method	Consistency (Intra) ↑	Consistency (Inter) ↓	F(1)-F(5) ↑	Sp ↑	Stability	C Acc ↑
Cars-196	Baseline	**83.45 ± 2.7**	27.26 ± 28.10	1.54 - 6.61	22.74	65.4	84.1
	ProtoPNet	45.70 ± 4.9	11.22±5.5	9.3-80.12	60.92	69.6	84.5
	PiPNet	42.40±17.5	17.4±2.4	9.43 - **93.15**	63.19	60.8	86.46
	ProtoViT	38.08±8.1	18.91±1.8	11.74 - 75.66	51.23	57.3	**91.84**
	MCPNet	48.91±14.6	9.17±3.2	14.78 - 88.02	60.42	64.26	80.15
	PCMNet (Ours)	57.84±3.1	**2.97 ± 1.37**	**59.33** - 91.73	57.45	**71.5**	90.15
CUB-200	Resnet50	57.38 ± 3.50	25.45 ± 23.82	2.43 - 5.73	24.14	60.3	81.2
	ProtoPNet	51.47 ± 5.8	10.16±9.2	9.36-75.50	79.51	63.6	81.45
	PiPNet	42.28±3.2	15.57±15.7	9.40 - 89.77	77.83	65.9	82.0
	ProtoViT	51.76±3.8	13.60±12.5	14.22 - 58.98	54.93	56.4	85.3
	MCPNet	52.94±5.9	8.11±5.0	24.60 - 67.30	50.33	58.4	80.1
	PCMNet (Ours)	**57.37±3.4**	**3.37 ± 4.2**	**54.34 - 89.84**	58.67	**68.5**	85.1

Table 1: Performance of PCMNet and state-of-the-art methods on the Stanford Cars and Birds datasets according to xAI metrics.

faithfulness (F(3)). However, excessive merging (e.g., Level 3 at 10%) leads to performance degradation, suggesting the importance of controlled prototype compression.

(b) Effect of Modules: To assess the impact of PCMNet components, we conduct an ablation study on the baseline PDiscoNet model without concept learning. The effect of using different types of concept vectors—prototypical and non-prototypical—is examined in Appendix B. Table 3 reports classification accuracy (Acc) and faithfulness (F(3)) on the Cars and Birds datasets. Adding the Marginal clustering center (MCC) loss improves accuracy and doubles faithfulness, highlighting the benefits of enforcing cluster-level consistency. Incorporating the Concept Mining Module (CMM) yields a major boost in faithfulness (from 4.9% to 55.3% on Birds), with minimal cost to accuracy. Combining both modules yields the best overall performance: 90.2% accuracy and 67.4% faithfulness on Cars, with only +0.4M parameters and +0.3G FLOPs. Compared to recent interpretable baselines like PipPNet, ProtoPNet, ProtoViT, and MCPNet, PCMNet achieves a better balance between accuracy, faithfulness, and efficiency, especially outperforming ProtoViT in F(3) (+22.8%) and requiring fewer FLOPs (17.4G vs. 24.8G). This confirms the complementary value of MCC and CMM in delivering interpretable and efficient visual recognition. The trade-off information is discussed in Appendix A of the extended version.

Threshold (%)	Level	d_c	C Acc (%)	F(3) (%)
0%	1	3092	90.2	67.4
	2	2473	85.7	65.7
	3	698	84.4	66.5
5%	1	2862	89.5	66.5
	2	1805	83.6	66.2
	3	645	80.5	65.7
10%	1	1657	86.2	67.9
	2	1263	82.7	66.9
	3	454	74.7	65.1

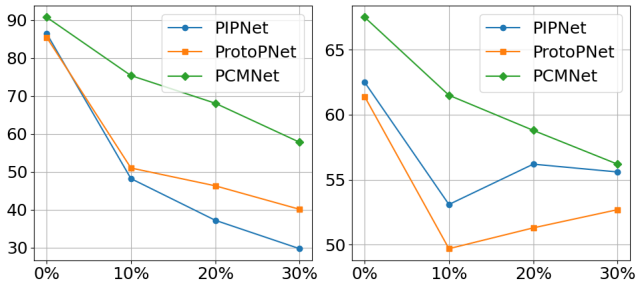
Table 2: Impact of hierarchical clustering thresholds and levels on the number of concepts (d_c), accuracy, and faithfulness (F(3)) on the Cars dataset. Thresholds expressed the percentage of the maximum pairwise centroid distance.

Settings	Cars		Birds		#Params	FLOPs
	Acc	F(3)	Acc	F(3)		
Baseline	81.2	5.4	82.3	4.9	27.1M	17.1G
Baseline + MCC	87.8	9.2	84.9	8.9	27.2M	17.1G
Baseline + CMM	85.8	58.6	82.5	55.3	27.4M	17.4G
PCMNet (Full)	90.2	**67.4**	85.1	**64.9**	27.5M	**17.4G**
PiPNet	86.4	62.7	82.0	58.6	25.1M	27.6G
ProtoPNet	84.5	38.6	81.4	31.7	28.9M	18.4G
ProtoViT	**91.8**	44.6	**85.1**	36.7	22.1M	24.8G
MCPNet	80.1	64.0	80.1	39.3	24.6M	**17.4G**

Table 3: Classification accuracy and faithfulness using different PCMNet modules.

Robustness to Occlusion

To evaluate PCMNet robustness under occlusion, we conducted controlled experiments by masking regions linked to the most activated concepts and re-evaluating the modified images with the model. This experiment aims to assess the stability of model predictions when key interpretable regions are removed. Specifically, we identify the center of the most activated concept—based on the model's response to clean images—using the concept representation (patch for PIPNet and ProtoPNet, mask for PCMNet), and then occlude a rectangular region centered around this point. We consider three levels of occlusion, masking 10%, 20%, and 30% of the image area. PCMNet's performance under these conditions is compared against two other interpretable baselines: ProtoPNet (Chen et al. 2019) and PIPNet (Nauta et al. 2023).



(a) Accuracy. (b) Faithfulness F(3)

Figure 3: Classification accuracy and faithfulness for a growing level of occlusion.

Fig. 3 (a) shows that PCMNet degrades more gradually under occlusion compared to ProtoPNet and PIPNet. This robustness comes from PCMNet's semantically rich and diverse part concepts, which capture complementary aspects of the object. Also, activated concepts from PCMNet preserve the faithfulness since they are extracted from not occluded regions as shown in Fig. 3 (b) and buttom of Fig. 4. By describing class-discriminative features from multiple perspectives, PCMNet enables more reliable predictions. In contrast, patch-based models depend heavily on fixed, localized cues and thus exhibit sharper performance drops.

Figure 4: Visualizations of activated concept on test samples with matching training samples for Stanford Cars and CUB.

Qualitative Results

To further illustrate the interpretability and robustness of PCMNet, we compare the activated concepts of our method with PIPNet under clean and occluded conditions, as visualized in Fig. 4. For each method, we select two activated concepts on test images and retrieve training set examples with the highest activation values for those concepts. This comparison shows several key advantages of PCMNet:

Broader Concept Coverage: PCMNet extracts more diverse and semantically rich concepts compared to PIPNet. It leverages attention-based masks instead of fixed patches to capture part-level semantics over a broader spatial context, leading to more varied and interpretable regions across different object parts and improving explanation quality.

Robustness to Occlusion: As shown at the bottom of Fig. 4, under occlusion, PIPNet often fails to highlight meaningful regions due to its reliance on limited regions. In contrast, PCMNet maintains interpretability by activating alternative, unoccluded parts that belong to the same or related concepts. This demonstrates the model's ability to generalize concept reasoning even when certain visual cues are missing. To better understand the behavior of the learned prototypes in PCMNet, we visualize the extracted part-level features using t-SNE (van der Maaten and Hinton 2008), as shown in Fig. 5. Each point corresponds to a feature from a localized part, colors indicating the prototype (i.e. concept) and marker shapes denoting the object class. This visualization represents several insights:

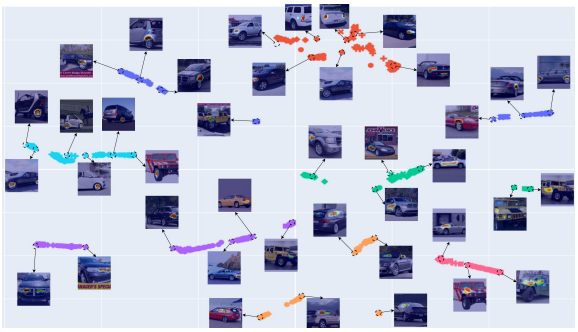


Figure 5: t-SNE visualization of concept-level part features extracted by PCMNet. Colors represent different prototypes, while shapes indicate object classes.

Disentanglement of Prototypes: Features assigned to different prototypes form well-separated clusters, indicating that PCMNet effectively disentangles part-based concepts. This shows that the Marginal Clustering Center loss and CMM promote diverse, semantically distinct parts, enhancing interpretability and occlusion robustness.

Multi-Class Concept Coverage: Many prototype clusters contain the same semantic parts from multiple object classes. This indicates PCMNet learns class-agnostic concepts such as lights, wheels, or windows, which are shared across different categories. Such generalization is crucial for explainability, as it facilitates human-aligned, semantically meaningful concepts that transcend dataset labels.

Intra-Class Concept Diversity: Interestingly, features from the same class are distributed across multiple prototype clusters. This highlights PCMNet's ability to capture *intra-class variation* through multiple part-based concepts. For instance, different views or structural variations of a car class may be assigned to distinct concepts (Backlight vs headlight), further enhancing the model's fine-grained interpretability based on human perception and explanation.

Conclusion

We presented PCMNet, a novel framework designed to bridge the gap between accuracy and interpretability in deep learning models. By mining part-prototypes from dynamic image regions, PCMNet improves upon existing patch-based prototype models by ensuring semantic coherence across instances. Our results show that PCMNet enhances faithfulness, concept stability, and explainability, outperforming existing interpretable models. Moreover, our occlusion-based robustness analysis confirmed that PCMNet remains stable under missing or distorted input regions, demonstrating its reliability in real-world scenarios. Unlike conventional post-hoc explainability methods, PCMNet allows for more human-aligned, concept-based reasoning, making deep learning decisions more transparent and interpretable.

Acknowledgments

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), and the Digital Research Alliance of Canada.

References

Alehdaghi, M.; Shamsolmoali, P.; Cruz, R. M.; and Granger, E. 2025. Bidirectional multi-step domain generalization for visible-infrared person re-identification. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 763–773.

Ayoobi, H.; Potyka, N.; Toni, F.; and DUMMY. 2025. ProtoArgNet: Interpretable image classification with super-prototypes and argumentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 1791–1799.

Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; and Samek, W. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7): e0130140.

Biederman, I. 1987. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2): 115.

Chai, J.; Zeng, H.; Li, A.; and Ngai, E. W. 2021. Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications*, 6: 100134.

Chakraborty, T.; Trehan, U.; Mallat, K.; and Dugelay, J.-L. 2022. Generalizing adversarial explanations with Grad-CAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 187–193.

Chan, C. S.; Kong, H.; Liang, G.; and DUMMY. 2022. A comparative study of faithfulness metrics for model interpretability methods. *arXiv preprint arXiv:2204.05514*.

Chefer, H.; Gur, S.; Wolf, L.; and DUMMY. 2021. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Chen, C.; Li, O.; Tao, D.; Barnett, A.; Rudin, C.; and Su, J. K. 2019. This looks like that: deep learning for interpretable image recognition. *Advances in neural information processing systems*, 32.

Dasgupta, S.; Frost, N.; Moshkovitz, M.; and DUMMY. 2022. Framework for evaluating faithfulness of local explanations. In *International Conference on Machine Learning*, 4794–4815.

Dreyer, M.; Achtibat, R.; Samek, W.; and Lapuschkin, S. 2024. Understanding the (Extra-)Ordinary: Validating Deep Model Decisions with Prototypical Concept-based Explanations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 3491–3501.

Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X.; et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, 226–231.

Fellous, J.-M.; Sapiro, G.; Rossi, A.; Mayberg, H.; and Ferrante, M. 2019. Explainable artificial intelligence for neuroscience: behavioral neurostimulation. *Frontiers in neuroscience*, 13: 1346.

He, H.; Zhu, L.; Zhang, X.; Zeng, S.; Chen, Q.; and Lu, Y. 2025. V2C-CBM: Building Concept Bottlenecks with Vision-to-Concept Tokenizer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 3401–3409.

Jiang, Y.; Zhao, X.; Wu, Y.; and Chaddad, A. 2025. A Knowledge Distillation-Based Approach to Enhance Transparency of Classifier Models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 17653–17661.

Kim, B.; Wattenberg, M.; Gilmer, J.; Cai, C.; Wexler, J.; Viegas, F.; et al. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, 2668–2677.

Kim, H.-S.; and Joe, I. 2022. An XAI method for convolutional neural networks in self-driving cars. *PLoS one*, 17(8): e0267282.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, 554–561.

Li, L.; Wang, B.; Verma, M.; Nakashima, Y.; Kawasaki, R.; and Nagahara, H. 2021. SCOUTER: Slot Attention-based Classifier for Explainable Image Recognition. In *IEEE International Conference on Computer Vision*.

Locatello, F.; Weissenborn, D.; Unterthiner, T.; Mahendran, A.; Heigold, G.; Uszkoreit, J.; and Kipf, T. 2020. Object-centric learning with slot attention. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Ma, C.; Donnelly, J.; Liu, W.; Vosoughi, S.; Rudin, C.; and Chen, C. 2024. Interpretable Image Classification with Adaptive Prototype-based Vision Transformers. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Markus, A. F.; Kors, J. A.; Rijnbeek, P. R.; and DUMMY. 2021. The role of explainability in creating trustworthy artificial intelligence for health care: a comprehensive survey of the terminology, design choices, and evaluation strategies. *Journal of biomedical informatics*, 113: 103655.

Monet, D.; Greff, K.; Van Steenkiste, S.; Locatello, F.; and Bachem, O. 2021. Object-centric learning with slot-based attention. In *International Conference on Learning Representations (ICLR)*.

Murtagh, F.; and Legendre, P. 2011. Ward's hierarchical clustering method: clustering criterion and agglomerative algorithm. *arXiv preprint arXiv:1111.6285*.

Nauta, M.; Schlötterer, J.; Van Keulen, M.; and Seifert, C. 2023. Pip-net: Patch-based intuitive prototypes for interpretable image classification. In *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2744–2753.

Nauta, M.; Van Bree, R.; Seifert, C.; and DUMMY. 2021. Neural prototype trees for interpretable fine-grained image recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14933–14943.

Oikarinen, T.; Das, S.; Nguyen, L. M.; and Weng, T.-W. 2023. Label-free concept bottleneck models. *arXiv preprint arXiv:2304.06129*.

Rao, S.; Mahajan, S.; Böhle, M.; and Schiele, B. 2024. Discover-then-name: Task-agnostic concept bottlenecks via automated concept discovery. In *European Conference on Computer Vision*, 444–461. Springer.

Rymarczyk, D.; Struski, Ł.; Tabor, J.; and Zieliński, B. 2021. Protopshare: Prototypical parts sharing for similarity discovery in interpretable image classification. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 1420–1430.

Saranya, A.; Subhashini, R.; DUMMY; and DUMMY. 2023. A systematic review of Explainable Artificial Intelligence models and applications: Recent developments and future trends. *Decision analytics journal*, 7: 100230.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*.

Srinivas, S.; and Fleuret, F. 2019. Full-Gradient Representation for Neural Network Visualization. In *Advances in Neural Information Processing Systems (NeurIPS)*.

van der Klis, R.; Alaniz, S.; Mancini, M.; Dantas, C. F.; Ienco, D.; Akata, Z.; and Marcos, D. 2023. PDiscoNet: Semantically consistent part discovery for fine-grained recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1866–1876.

van der Maaten, L.; and Hinton, G. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86): 2579–2605.

Vilone, G.; Longo, L.; DUMMY; and DUMMY. 2020. Explainable artificial intelligence: a systematic review. *arXiv preprint arXiv:2006.00093*.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset.

Wang, B.-S.; Wang, C.-Y.; and Chiu, W.-C. 2024. MCP-Net: An Interpretable Classifier via Multi-Level Concept Prototypes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10885–10894.

Wang, H.; Wang, Z.; Du, M.; Yang, F.; Zhang, Z.; Ding, S.; Mardziel, P.; and Hu, X. 2020. Score-CAM: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 24–25.

Wong, E.; Santurkar, S.; and Madry, A. 2021. Leveraging sparse linear layers for debuggable deep networks. In *International Conference on Machine Learning*, 11205–11216.

Xie, Y.; Zeng, Z.; Zhang, H.; Ding, Y.; Wang, Y.; Wang, Z.; Chen, B.; and Liu, H. 2025. Discovering Fine-Grained Visual-Concept Relations by Disentangled Optimal Transport Concept Bottleneck Models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 30199–30209.

Yuksekgonul, M.; Wang, M.; and Zou, J. 2023. Post-hoc Concept Bottleneck Models. In *The Eleventh International Conference on Learning Representations*.

Zhu, Z.; Jin, Z.; Zhang, J.; and Chen, H. 2024. Enhancing model interpretability with local attribution over global exploration. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 5347–5355.